

# ***PROTOGEN<sup>TM</sup>***

**C and ObjectWindows Code Generator**



*ProtoGen*™

---

## User's Guide

ProtoView Development Corporation  
353 Georges Road  
Dayton, N.J. 08810  
(908) 329-8588

Information in this document is subject to change without notice and does not represent a commitment on the part of ProtoView Development Corporation. The software described in this document is furnished under the provisions of a license agreement. The purchaser, and only the purchaser, may make one copy of the development software for backup purposes. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including but not limited to photocopying and recording, for any purpose without the written consent of ProtoView Development Corporation.

© Copyright ProtoView Development Corporation, 1992. All rights reserved.

© COMMDLG.DLL is copyright 1991 by Microsoft Corporation.

ProtoView and the ProtoView logo are registered trademarks, and ProtoGen and ViewPaint are trademarks of ProtoView Development Corporation.

Borland and Borland C++ are trademarks of Borland International Corporation.

Microsoft, Microsoft Windows, MS DOS are registered trademarks of Microsoft Corporation.

<b>Introduction</b>	1	<b>WorkingDirectory</b>	21
ProtoGen benefits	1	Header File	22
ProtoGen features	2	Exit	23
Hardware and software requirements	2		
<b>Chapter 1 Getting started</b>	3	<b>Chapter 4 Edit menu</b>	25
Overview	3	Design Menu	25
Installing ProtoGen	4	.MNU files	28
Starting ProtoGen	5	.RC file	28
General	5	Top menu bar	29
Mnemonics	6	File	29
Accelerators	6	Edit	30
Other features	6	Menu	30
The ProtoGen Application Generator	7	Add Menu Item	31
ProtoGen main menu bar	7	Add Menu Item to Action Bar	32
		Menu Designer buttons	32
		Examples	33
		Change example	33
		Insert example	34
		Cascade example	34
		Separator example	34
		Item State	34
		Menu Options	35
		Accelerator keys	35
		Exiting Menu Design	35
		Main Window Dialog	36
		Application Icon	36
		Window Caption	37
		Window Colors	37
		Remove Window Colors	38
		Edit Resources	38
<b>Chapter 2 Tutorial</b>	9	<b>Chapter 5 Dialog boxes</b>	41
Opening a .MNU file	9	DLG files	41
Changing a menu item	12	Linking dialog boxes to menu items	42
Inserting menu items	12	One .DLG linked to multiple menus	44
Adding a cascade	13	Step-by-step linking	44
Adding a separator	13		
Adding an accelerator	14		
Linking dialog boxes	14		
Verifying the link	16		
Attaching subdialog boxes	16		
Verifying links	17		
Removing dialog boxes	17		
Changing modality	18		
Naming and coloring the application	18		
<b>Chapter 3 File menu</b>	19		
File Menu selections	19		
New	19		
Open	20		
Save	20		
SaveAs	21		

Unlinking dialog boxes from menu items .....	44
Remove .....	45
Linking nested dialog boxes .....	45
Escaping from dialog box add or edit mode .....	47
Linking dialog boxes to the main window .....	47
<b>Chapter 6 Application menu</b> .....	49
Test .....	49
Generate .....	50

Generate Options .....	51
ANSI C Generation .....	51
Borland C++ Generation .....	51
Debugging Information .....	51
Use Borland Controls .....	52
Static Linking .....	52
Compile .....	53
Execute .....	54
<b>Chapter 7 Tools and other topics</b> .....	55
Hints on using ProtoGen .....	56
<b>Index</b> .....	57

1.1: Accelerator keys .....6

1.1: Application Generator screen . . . . .	5	4.5: Save As . . . . .	30
2.1: File menu . . . . .	10	4.6: Separator function . . . . .	33
2.2: Application Generation screen . . . . .	10	4.7: Application Generator . . . . .	36
2.3: Open Menu Resource File screen . . . . .	11	4.8: Select Application Icon . . . . .	37
2.4: Menu Designer . . . . .	11	4.9: Edit Application Window Caption . . . . .	37
2.5: Separator . . . . .	14	4.10: Color selection . . . . .	38
2.6: File menu . . . . .	15	4.11: Edit Application Window Caption . . . . .	39
2.7: File menu . . . . .	15	5.1: Add Application Dialog . . . . .	43
3.1: ProtoGen Open dialog box . . . . .	20	5.2: Edit Application Dialog . . . . .	44
3.2: ProtoGen SaveAs dialog box . . . . .	21	5.3: Nested dialog boxes . . . . .	46
3.3: ProtoGen Working directory dialog box . . . . .	22	5.4: Edit Application Dialog . . . . .	46
3.4: ProtoGen Header File Dialog box . . . . .	22	5.5: Main Window Dialog . . . . .	47
3.5: File menu . . . . .	23	6.1: Application menu . . . . .	49
4.1: Edit menu . . . . .	25	6.2: Set Compiler Options . . . . .	50
4.2: Application Generator . . . . .	26	6.3: Compile dialog box . . . . .	53
4.3: Menu Designer screen . . . . .	27	6.4: Execute dialog box . . . . .	54
4.4: Open template file . . . . .	29	7.1: Update Tools Menu . . . . .	55

This document describes how to use ProtoGen for creating menus, linking and unlinking dialog boxes, generating final code in C++ and ANSI C, and developing Windows application prototypes. ProtoGen is itself a Windows application. It works with the ProtoView Screen Management Facility, with Windows alone, or in combination with both environments. Companion products ProtoView and the ProtoView DataTable offer a comprehensive screen management facility for Windows with sophisticated access to databases for browsing and updating.

Intermediate output lets you construct, test, and modify an application prototype before expensive development on the final code is started. The final output of ProtoGen can be compiled and linked with Borland International's ObjectWindows for C++. ProtoGen produces a fast, bug free, and modular source code that can be modified and extended easily.

ProtoGen and this document are intended for use by beginning and experienced Windows programmers. To start using ProtoGen right away, see Chapter 2 for a step-by-step tutorial. See Chapter 5 for more information about dialog boxes and how to link them to menu items.

## ProtoGen benefits

---

ProtoGen helps numerous people involved in the development cycle of a Windows application. Typically, as the first step in application development, the end user tells the analyst how the application should work. Then, with ProtoGen, the analyst produces a user interface compatible with the other code that makes up a Windows application. Finally, using ProtoGen and either Borland's Resource Workshop or ProtoView's ViewPaint



(available separately), the user interface screens are changed as features are added and modified.

Since ProtoGen is an effective design tool at various levels of application development, people with diverse skills can communicate their ideas in a visual and nontechnical manner. Because ProtoGen simplifies such tasks as developing menus and accelerator (shortcut) keys, linking dialog boxes to menus, and generating code for the framework of a Windows application, it frees the experienced developer to work on more critical aspects of the application.

## ProtoGen features

---

ProtoGen offers the following features:

- A graphically oriented user-interface design tool for quickly designing menus for Windows applications.
- A code generator compatible with ANSI C and Borland's ObjectWindows for C++ that integrates many levels of dialog boxes, created with another software package, into your applications.
- An application layout and prototyping tool.
- A tester for trying out the interface before finalizing code.

## Hardware and software requirements

---

ProtoGen minimum requirements are Windows Version 3.0 or later in Standard or 386 Enhanced mode, at least 2.0 megabytes of RAM, and 3 MB of free space on the hard disk. A mouse or other pointing device is optional.

To create executable versions of your prototypes using ANSI C, you need the Borland C++ version 3.0 compiler or another compiler capable of compiling for Windows.

To create executable versions of your prototypes using C++, you need the Borland C++ version 3.0 & Application Frameworks, which includes ObjectWindows.

# Getting started

## Overview

---

Application development under Windows is often complex, difficult, and intimidating, even for experienced C and C++ programmers. The following section tells you how to use ProtoGen's prototyping capability to make the creation of Windows applications easy—as easy as using Windows applications. Prototyping lets a developer put together a working demonstration of an application, show it to the user, make needed modifications, show it again, and repeat the process until the prototype becomes the working program.

The following list describes the basic steps involved in using ProtoGen:



1. Tell ProtoGen the location of the application header file. See Chapter 3.
2. Create a menu for your application from scratch or edit an existing menu contained in an .RC or .MNU file. Design menus with submenus or popups, grayed or disabled items, accelerator keys, separators, and menu bar breaks in any combination and order. ProtoGen lets you move, change, add, insert, or remove menu items easily.
3. Link dialog boxes contained in existing .DLG files to menu items. (The .DLG files are created using either the Windows SDK, ProtoView's ViewPaint, Borland's Resource Workshop, or

similar software. See Chapter 5 for brief instructions on creating .DLG files with Resource Workshop.

4. Specify the icon and colors of the top level window.
5. Specify a caption for the top level window.
6. Attach subdialog windows to dialog buttons that don't already have a dialog box attached.
7. Change the attributes of the existing, attached dialog box, by simply pressing *F2* or right clicking the mouse when that dialog box has the focus.
8. Specify the main window of the application as a dialog box or view window. When you make the main window a dialog box, ProtoGen loads and displays the dialog box resource while the application is prototyped. Subdialog boxes are attached to controls on the main window just as if they were dialog boxes.
9. During the application design and prototyping stage, interactively test the current screen flow before generating code. The prototype can be shown to end users to get additional feedback and comments concerning the design and flow of the application.
10. Generate C or C++ code ready to compile and execute for the entire interface, including the header file, resource file, module definition file, make file, and linker response file. Any previous versions of modules containing custom code are regenerated with the added code intact. The code generator produces files for the main window and each of the dialog boxes contained in the application.
11. Finally, write the C or C++ code to create the functionality portion of your application, and connect it to the ProtoGen-generated code.

## Installing ProtoGen

---

Before installing ProtoGen, read the licensing agreement included in the product package, and make a backup copy of the installation disk. Perform the following steps to install ProtoGen:

1. Insert the disk into either drive A or B.
2. Make either drive A or B the current drive.
3. Enter the following command: `win install`

4. The installation dialog box is displayed with install and cancel buttons.
5. Click Install.
6. Enter the drive and path to install ProtoGen.

After you've loaded the disk contents, a dialog box shows the current AUTOEXEC.BAT file and proposed changes.

7. Edit the box with the proposed changes or click OK to update the AUTOEXEC.BAT file. Select Cancel to leave the file unchanged.

The PATH variable is updated to include the location of ProtoGen. Remember to run the AUTOEXEC.BAT for the new path setting to take effect.

8. Add the program icon.

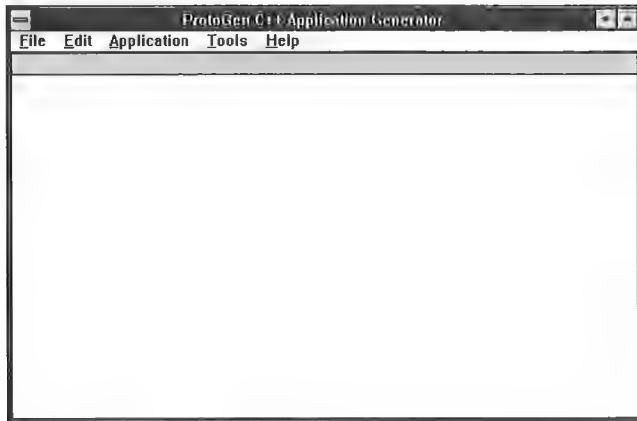
*Before attempting to edit the AUTOEXEC.BAT, click on the edit box to turn off text highlighting. Use CTRL ENTER instead of ENTER for multiple lines.*

## Starting ProtoGen

---

Double-click on the ProtoGen icon, located in a program group of the Windows Program Manager. The ProtoGen Application Generator screen is displayed.

Figure 1.1  
Application Generator  
screen



### General

---

Many ProtoGen operations are familiar to Windows users.

**Mnemonics** A mnemonic is an underlined letter in the name of a menu or menu item. For example, “F” is the mnemonic for the File menu on the ProtoGen Application Generator screen. See the “ProtoGen main menu bar” section, which shows mnemonics for menu items on ProtoGen’s main menu.

While using ProtoGen in test or prototype mode, the main menu mnemonics aren’t available until you press *F4* to switch out of test or prototype mode.

Typing the mnemonic selects the menu just as if you clicked on it.

**Accelerators** Accelerator keys are actually key sequences that let you quickly access menu functions. Instead of clicking on a menu and clicking on a menu item, you type a key sequence from the main window.

ProtoGen has the following accelerator keys:

Table 1.1  
Accelerator keys

Menu	Key sequence
<b>Main Menu:</b>	
File   New	<i>Ctrl F2</i>
File   Open	<i>Ctrl F3</i>
File   Save	<i>Ctrl F4</i>
File   Exit	<i>Alt F4</i>
Edit   Design Menu	<i>Ctrl M</i>
Application   Test	<i>Ctrl T</i>
Application   Generate	<i>Ctrl G</i>
<b>Menu Designer:</b>	
File   New	<i>Ctrl F2</i>
File   Open	<i>Ctrl F3</i>
File   Save	<i>Ctrl F4</i>
Edit   Cut	<i>Shift Del</i>
Edit   Copy	<i>Ctrl Ins</i>
Edit   Paste	<i>Shift Ins</i>
Menu   Add Menu Item	<i>Ctrl I</i>
Menu   Add Item to Action Bar	<i>Ctrl B</i>

**Other features** Grayed-out (dimmed) menu items and buttons are unavailable in the current mode.

Clicking on the upper left corner of ProtoGen’s windows displays the System menu. See your Windows documentation for information on the system menu and other basic Windows information, such as how to use maximize and minimize buttons.

This document describes only one way, the mouse method, of working with ProtoGen; however, ProtoGen menus and dialog boxes offer several methods of operation: keyboard shortcuts, accelerator keys, or mouse point-and-click methods. For example, open popup menus by typing the mnemonic in the menu's name or by clicking on the menu name. Select menu items by clicking on them or typing the mnemonic in the menu item's name.

The accelerator key sequence, if there is one, is displayed to the right of the menu item. Dialog boxes have buttons usually labeled OK and CANCEL. After entering data into the dialog box window, click on OK or press *Enter*. The CANCEL button escapes from the dialog box without changing anything.

## The ProtoGen Application Generator

---

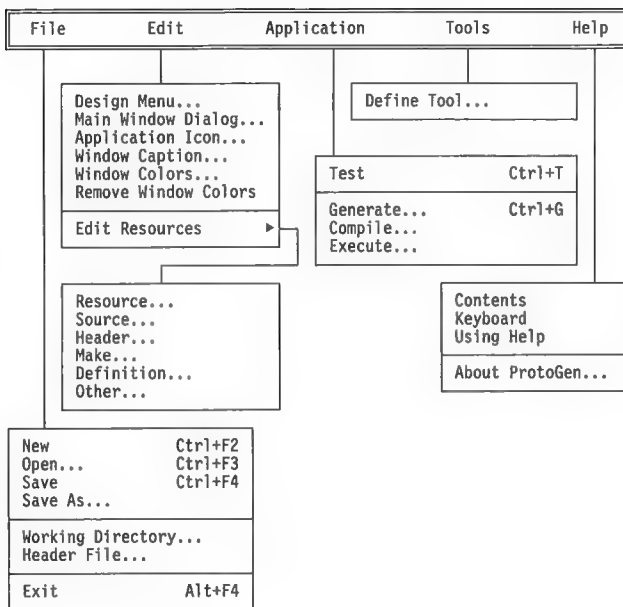
The ProtoGen Application Generator screen, shown in the previous figure, contains two menu bars: the upper main menu bar with File, Edit, Application, Tools, and Help popup menus, and the lower Application Prototype title bar, which initially displays "Application Caption."

## ProtoGen main menu bar

---

The following figure shows the main menu bar and its associated menus. The accelerator keys (shown on the right-hand side of the menu item) are for accessing ProtoGen functions by the shortcut method. The menus are discussed in detail in subsequent sections.

*If you opened files in ProtoGen previously, the names of the last four .PVA files are displayed on the File menu.*



## *Tutorial*

This chapter presents an example of how to perform the critical steps in creating an application prototype with ProtoGen. You open an existing .MNU file, edit it, observe changes on the prototype menu, and link dialog boxes to the prototype menu.

### Opening a .MNU file

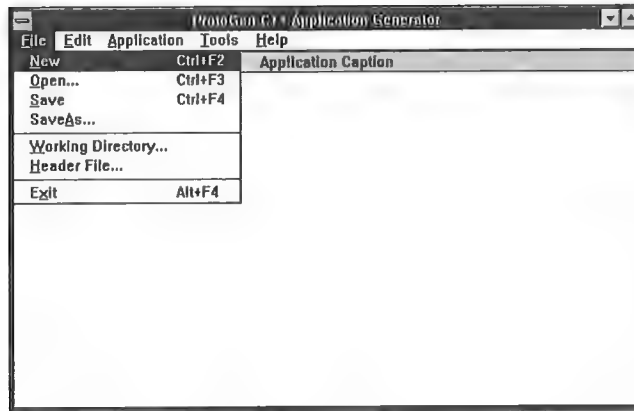
---

Assuming you've just installed ProtoGen and have the Windows Program Manager displayed, perform the following steps to open the .MNU file:

1. Double-click on the ProtoGen icon to invoke ProtoGen.
2. Select File | New from the following ProtoGen main menu.

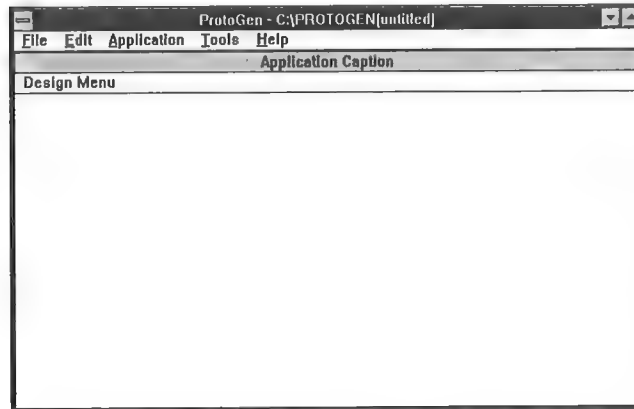


Figure 2.1  
File menu



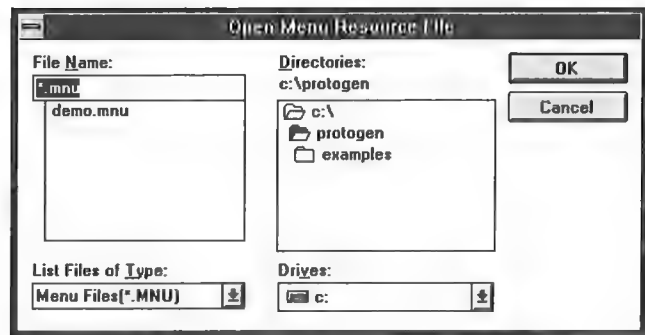
Selecting File | New in the last step displays the following screen:

Figure 2.2  
Application Generation  
screen



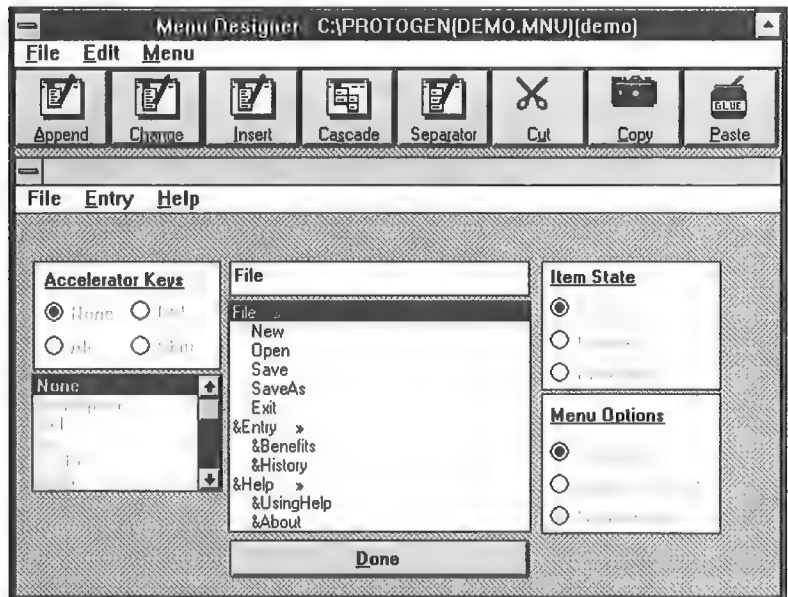
3. Click Design Menu. The Menu Designer screen is displayed.
4. Select File | Open from the Menu Designer menu bar to display the following screen:

Figure 2.3  
Open Menu Resource File  
screen



5. Select the EXAMPLES directory from the Directories list box, and click OK.
6. Select DEMO.MNU from the File Name list box, and click OK. You're asked if you want to copy the menu resource file to the working directory.
7. Click OK. The Menu Designer screen list box shows the contents of DEMO.MNU:

Figure 2.4  
Menu Designer



## Changing a menu item

---

Look at File, Entry, and Help on the prototype bar of the menu designer. The “E” in Entry and “H” in Help are underlined to indicate the mnemonic, but File has no mnemonic. The File menu contains New, Open, Save, SaveAs, and Exit items and none have mnemonic indicators. The Entry and Help menus, on the other hand, have mnemonics indicated for the first letter of each of their menu items. For example, Benefits on the Entry menu.

Pull down each menu, and inspect the menu items on it. Notice that menu items in the list box contain ampersands before the mnemonic letters.

The next few steps add a mnemonic, “F” to the File menu name by inserting an ampersand before File >> on the first line of DEMO.MNU in the list box:

1. Focus on the single-line entry window above the list box, which contains the line you want to edit; in this case it’s File. If the single-line entry window doesn’t contain the line you want to edit, focus on it in the list box, and move to the edit box.
2. Move the focus to the beginning of the word and type an ampersand.
3. Press Enter.  
The list box and prototype menu bar are updated with an ampersand and a mnemonic, respectively.
4. Select File | New from the prototype menu. New is displayed in the single-line entry window.
5. Repeat steps 1-3 to add the mnemonic to New.

Add mnemonics to Open, Save, SaveAs, and Exit menu items. The ampersand goes in front of the first letter for most menu items. The ampersand should be placed in front of the “A” in SaveAs and in front of the “x” in Exit.

## Inserting menu items

---

Add a popup menu called Edit between the Entry and Help menus by performing the following steps:

1. Select the line containing “&Help >>” from the list box.

2. Press Enter.
3. Type "&Edit".
4. Click the Insert button.

An Edit menu is inserted in between Entry and Help menus on the prototype menu bar, and "&Edit" is inserted before "&Help" in the list box.

## Adding a cascade

---

Add a cascaded menu to the Edit menu by clicking the Cascade button. Menu items with template names Item1, Item2, and Item3 are added to the list box and to the prototype Edit menu.

Change the template names to Employee, Benefits, and Medical by selecting the template text to edit from the list box and moving to the single-line entry box edit the names.

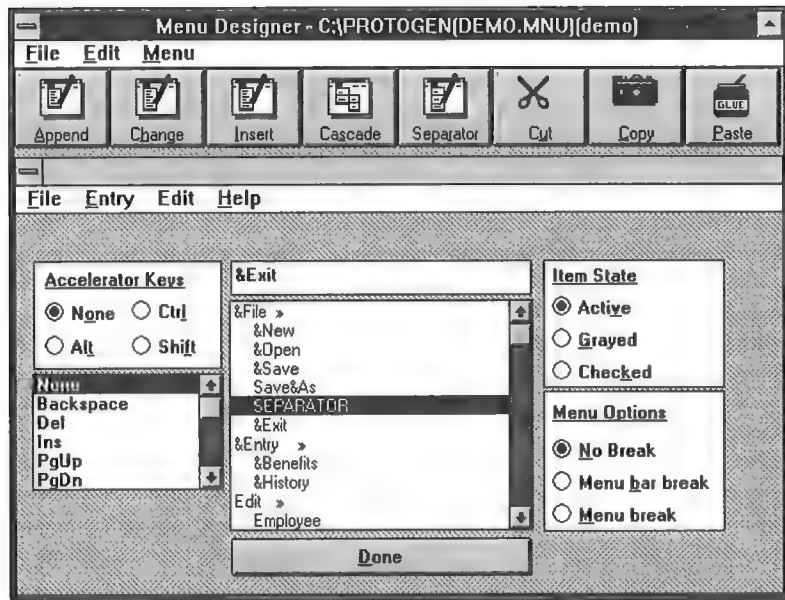
## Adding a separator

---

Add a separator between the SaveAs and Exit items on the File menu by selecting Exit and clicking the Separator button.

"SEPARATOR" appears in the list box, as shown in the following figure.

Figure 2.5  
Separator



Inspect the File menu on the prototype bar to see the effect of adding the separator.

## Adding an accelerator

Perform the following steps to add the menu accelerator, *ALT+S*, to the Save item on the File menu.

1. Select File | Save from the prototype menu.
2. Select Alt in the Accelerator Keys window.
3. Select S in the Accelerator Keys scroll box.
4. Press Enter.

Inspect the File menu on the prototype menu bar and verify that *Alt+S* is displayed adjacent to Save.

## Linking dialog boxes

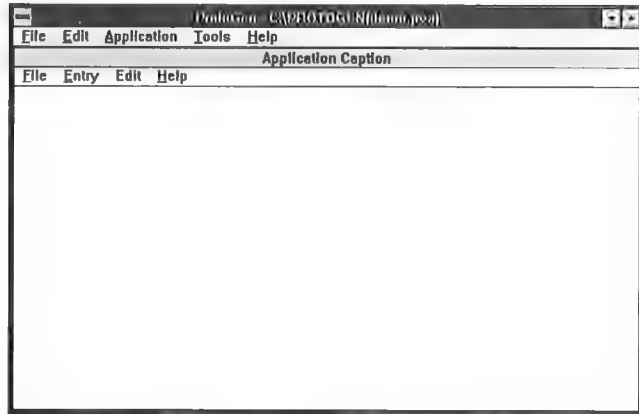
The following steps link dialog boxes to menu items displayed on the main ProtoGen screen below "Application Caption":

1. Exit the menu design mode by selecting File | Save followed by File | Done from the Menu Designer menu, not from the prototype menu.

The ProtoGen main window is displayed again, but now instead of “Design Menu” the menu you created (File Entry Edit Help) from the DEMO.MNU template appears on the prototype menu bar below Application Caption.

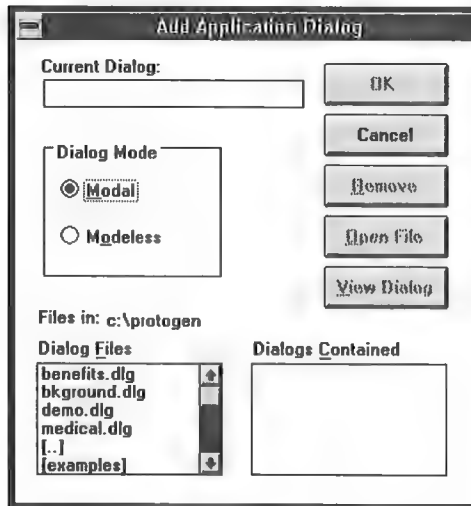
2. Select File | New from your prototype menu, which is the lower menu on the following figure:

Figure 2.6  
File menu



The Add Application Dialog screen is displayed with a scroll list of drives and directories in the bottom, left-hand window.

Figure 2.7  
File menu



3. Go to the directory, EXAMPLES by double-clicking on [examples].
4. Select DEMO.DLG.
5. Press Enter or click on the Open File button  
DLG resources contained in DEMO.DLG are displayed in the bottom, right-hand window.
6. Select the dialog box to be attached; in this case DLG\_FILENEW, from the Dialogs Contained list.
7. Press Enter.
8. Click OK to the question, "Copy dialog resource file to Working Directory?"  
The selected dialog box, labeled "New", is attached to the File | New menu item.

## Verifying the link

---

To verify the link to the dialog box, select File | New. The dialog box "New" displayed.

To close the "New" dialog box, double click on the client area of the New dialog box. Alternatively, this particular dialog box could be closed by selecting Close from the New's control-menu box, but some dialog boxes won't offer this option.

## Attaching subdialog boxes

---

Dialog boxes attached to other dialog boxes are called subdialog boxes. Perform the following steps to attach a subdialog box to the "New" dialog box on File | New from the prototype menu.

1. Select File | New.  
The "New" dialog box is displayed.
2. Attach a subdialog box to the Okay button of "New" by clicking on the Okay button.  
The Add Application Dialog screen is displayed.
3. Go to the directory, EXAMPLES, by double-clicking on [examples].
4. Select DEMO.DLG.

5. Press Enter or click the Open File button.
6. Select DLG\_SETUP from the Dialogs Contained window.
7. Click OK.  
The Add Application Dialog screen disappears, and the "Setup" screen is in view.
8. Double-click on the client area of the "Setup" screen to escape, getting back to the parent ("New") dialog box.
9. Double-click on the client area of the "New" screen to get back to the main ProtoGen window.

## Verifying links

---

Verify that the subdialog box was linked correctly by performing the following steps:

1. Select File | New, and click the Okay button on the "New" (parent) dialog box.  
The subdialog box called "Setup" is displayed.
2. Double-click on the client area of the subdialog box to get back to the parent dialog box.
3. Double-click on the client area of the parent dialog box to get back to ProtoGen's main window.

## Removing dialog boxes

---

To unlink the subdialog box, "Setup", from the application, perform the following steps:

1. Go to the ProtoGen main window, select Edit | Employee to display the "New" dialog box, and click Okay to display the subdialog box, "Setup".
2. Click the right mouse button or press *F2*.  
The Edit Application Dialog screen is displayed.
3. Click the Remove button.
4. Click the Yes button when asked if you want to delete the dialog box.  
The subdialog box is unlinked and the parent dialog box is in view again.



5. Double-click on the client area of the parent dialog box to escape to ProtoGen's main menu.

## Changing modality

---

Change the modality of a dialog box by repeating steps in the previous section to get the Edit Application Dialog screen. When the Edit Application Dialog screen is displayed, select modal or modeless.

## Naming and coloring the application

---

See Chapter 4 for instructions on performing the simple operations involved in naming and coloring an application and selecting an icon for it.

## File menu

This chapter describes selections on the File menu, which create or access application prototype files.

File	
New	Ctrl+F2
Open...	Ctrl+F3
Save	Ctrl+F4
Save As...	
Working Directory...	
Header File...	
Exit	Alt+F4

### File Menu selections

---

The File Menu lets you open and close .PVA files (defined later), save prototype menu data, tell ProtoGen where to find files, and exit from ProtoGen. The following sections describe each menu selection in detail.

#### New

---

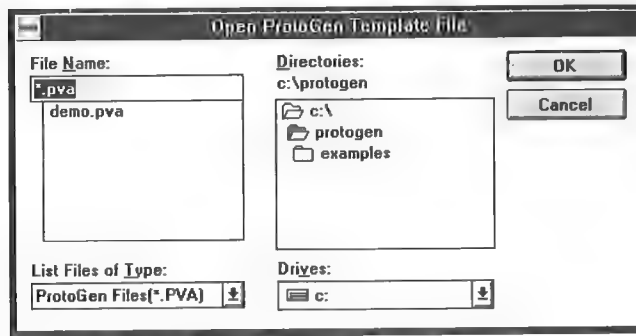
File | New begins a new application prototype as opposed to a prototype created previously. If any files are open in ProtoGen you'll be reminded to save them.

---

## Open

File | Open accesses existing application prototype template files, .PVA files. The following screen is displayed:

Figure 3.1  
ProtoGen Open dialog box



Click in the Drives scroll box to access the drive you want. Click in the Directories box followed by OK to access the directory you want. Click in the file box to access the file you want to open, or type the File Name in the window above the list box.

Upon opening an application (.PVA file), ProtoGen displays information from the menu template (.MNU file) or the resource script (.RC file) associated with the .PVA file, and creates the prototype menu.

A .PVA file is like a table of contents. It doesn't contain any actual data. It gives ProtoGen the following information:

- the type of application to generate (C or C++)
- the names of the header (.H), resource (.RC), module definition (.DEF), make or project (.MAK), and icon (.ICO) files.
- the name of the caption of the prototype
- the color of the client area of the prototype
- additional library directories

---

## Save

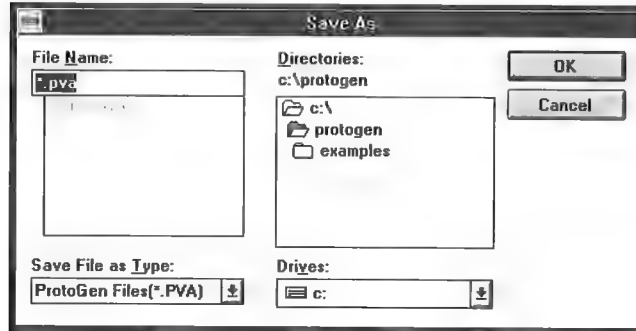
File | Save stores application prototype information including form, attributes, and the menu description file (.MNU or .RC file) to a disk file in the working directory. For new applications, ProtoGen prompts you for a file name before performing the operation. Enter a file name without an extension. File | Save saves

the prototype information to the .PVA file and saves the .MNU file according to the application's name. See Chapter 4 for more information about .MNU and .RC files.

## SaveAs

File | SaveAs stores application prototype information under a different name, and allows you to change the file names of both the .PVA and the .RC files. See the previous instructions for File | New. File | SaveAs works similarly. Using SaveAs with template menus (.MNU files) lets you keep the template file intact. ProtoGen sets the working directory according to the directory specified with the SaveAs command is used.

Figure 3.2  
ProtoGen SaveAs dialog box



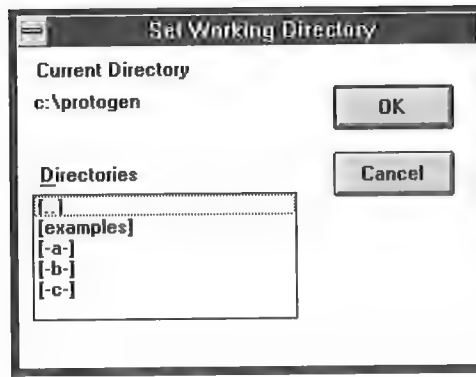
## WorkingDirectory

File | WorkingDirectory specifies the file directory where ProtoGen writes and overwrites the .PVA and supported files. To change the working directory, double-click on the directory or drive letter, and click OK. By default, the current directory is the working directory. Selecting an application in another directory changes the working directory to that directory.



To prevent overwriting files inadvertently, check which directory you're in. Look at either the Application Caption section of ProtoGen's main menu bar or the Current Directory window in the WorkingDirectory dialog box. Both display the name of the working directory unless you've just invoked ProtoGen and haven't opened a file yet.

Figure 3.3  
ProtoGen Working directory  
dialog box

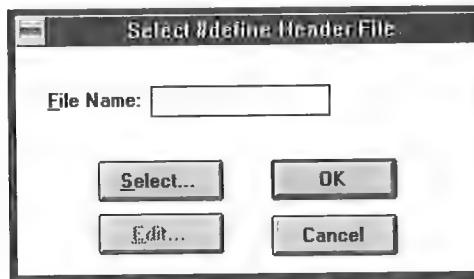


## Header File

File | Header File displays the following dialog box, which performs several functions.

Figure 3.4  
ProtoGen Header File Dialog  
box

**Important:** ProtoGen can't generate code without knowing the location of the header file.



The Open button on the Open Disk file dialog opens a file, but won't automatically invoke an editor.

The Select button brings up the Open Disk file dialog box that lets you select a .H file in any directory. You can put the file name in the File name input box or change to another directory using the directory box. Next you can select the file using the Files list box.

The header file contains #define preprocessor directives used in generating the application, like those in the following example:

```
#define Id_Splash 102
#define id_help 210
#define ID_ScribbleDirectory 219
#define id_TextDirectory 220
#define id_GraphicDirectory 221
#define id_FList 101
#define id_DList 102
#define id_Text 205
#define id_Scribble 206
```

```
#define id_Graph      207
#define id_FName      100
```

The Edit button on the Header File Dialog box displays the .H file in an editor, similar to the NotePad. It has popup menus for saving the new file and for cutting and pasting changes to the .H file, which must be less than 32K.

## Exit

---

File | Exit shuts down ProtoGen, asking you to save work in progress and saving current environment settings, such as workfile name, compiler model and size, and code generation options, and updating the .PVA and .MNU files.

The first time you work with ProtoGen, Exit is the last choice on the File menu, but in subsequent sessions ProtoGen displays the file names of the last four application prototypes created at the bottom of the menu.

Figure 3.5  
File menu

*A .PVA file can be selected from the menu with the mouse or by typing the underlined numeral.*

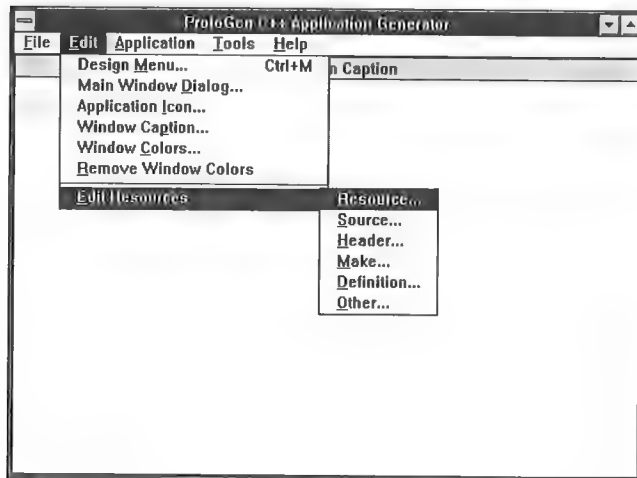
File	
New	00001.P
Open...	Ctrl+F3
Save	Ctrl+F4
SaveAs...	
Working Directory...	
Header File...	
Exit	Alt+F4
1 C:\PROTOGEN\demo.pva	



## *Edit menu*

This chapter describes each entry on the Edit menu in more detail. The Edit menu accesses ProtoGen's most important functions. The following figure shows items on the Edit menu.

Figure 4.1  
Edit menu



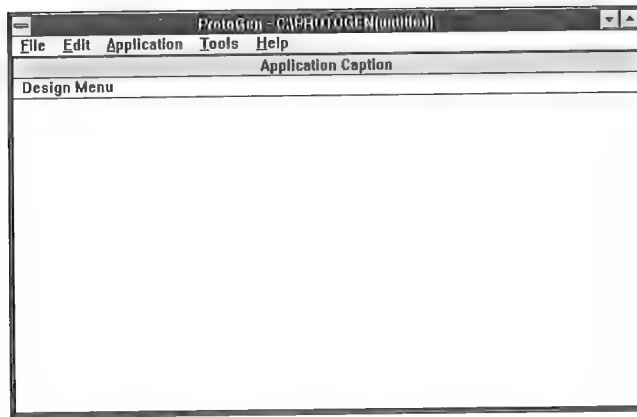
## Design Menu

The Design Menu displays controls for creating the prototype menu. From this menu, the Design Menu functions are invoked, and the following Menu Designer controls are displayed:



Figure 4.2  
Application Generator

*Design Menu appears in the  
bar under the Application  
Caption.*



After selecting File | New, the title bar of the Application Generator screen displays "Application Caption" in the center of the bar.

To alter and update the application prototype menu with the ProtoGen Menu Designer, invoke the Menu Designer in one of the following ways:

- After selecting File | New, click on "Design Menu" in the menu bar of the Application Generator, which is the lowest menu bar on the screen.

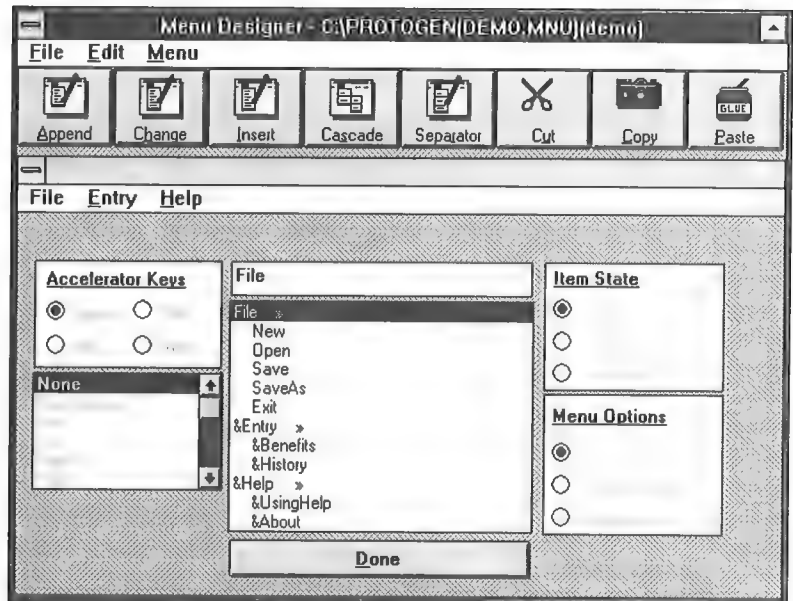
Select Design Menu from the Edit menu on the main menu.

The following Menu Designer window is displayed:

Figure 4.3  
Menu Designer screen

*In this example, the prototype menu contains File, Entry, and Help menus.*

*The single-line entry box is located above the list box in the lower central portion of the menu.*



*Alt+F6 or the mouse toggles between the Menu Designer menu bar and the top-level menu bar.*

The Menu Designer lets you create a specialized resource script, which is stored in a .MNU file. The Menu Designer creates identifiers and connects accelerator keys to menu items. See Chapter 1 for more information on Accelerator keys.

The following steps briefly describe how to design a menu from scratch:

1. Start the Menu Designer, using one of the following methods:
  - Select File | New followed by Edit | Design from the ProtoGen main menu bar (ProtoGen names the menu file the same as the .PVA file, but with a .MNU extension).
  - Select File | New from the Menu Designer menu, and enter a new menu resource name.
  - Select File | Open and select from a list of the .MNU files in the working directory or select a .RC file. To see a list of .RC files in the working directory, enter \*.rc in the File Name window.

2. Make a new menu using the controls on the Edit and Menu menus, and using the other Menu Designer controls, which are described later.
3. Click Done below the list box.
4. Select File | Save, and enter the name of the application if required.

---

## .MNU files

Menu template files are ASCII text files with the extension .MNU. Because they are reusable in different applications, the template files free you from redesigning the menu each time. Like resource script (.RC files), .MNU files contain the following data:

- a menu resource—an RC script that describes the structure of a menu
- an optional accelerator table—shortcut definitions for Accelerator keys.

As you edit a .RC file, .MNU template file, or create a file from scratch, the display in the Menu Designer list box represents the contents of the file, while the Menu Designer menu bar shows the prototype menu.

To navigate from the list box in the center of the screen to the menu bar and vice versa, click on menu items in either area. When you highlight a line in one area, the line is addressed in the other area automatically. Associated Accelerator keys are highlighted as well as menu states. Experiment and you'll quickly appreciate how informative the Menu Designer is.

---

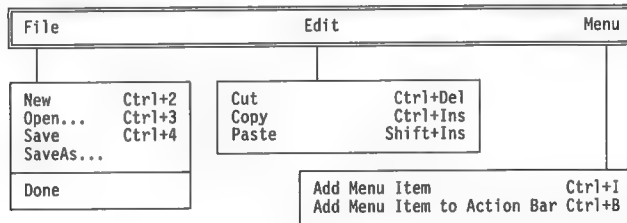
## .RC file

The following example shows how a .RC file appears in the list box of the Menu Designer:

```
&File >>
    &Load
    &Save
    &Print
    SEPARATOR
    &Exit
&Setup >>
    &Games
    &Stats
```

## Top menu bar

The top menu bar of the Menu Designer screen has menus located above the control buttons offering the following selections:

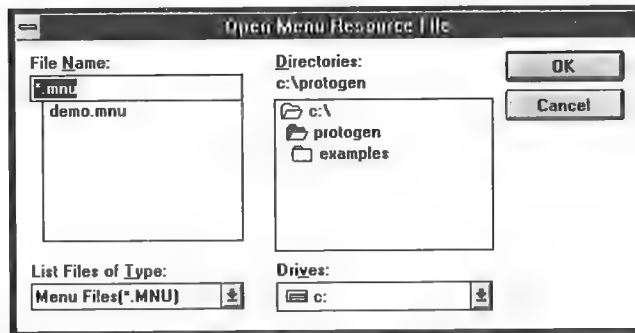


## File

Select File | New to start a new application. A dialog with an input box is displayed. Enter the resource name, which doesn't have to be the same as the resource file name. The resource name should be an acceptable name for a .MNU file.

Select File | Open to open an existing .MNU or .RC file. The following dialog box is displayed:

Figure 4.4  
Open template file  
*Click on the drive and file  
you want, or type the name  
of the file in the window  
under File Name.*

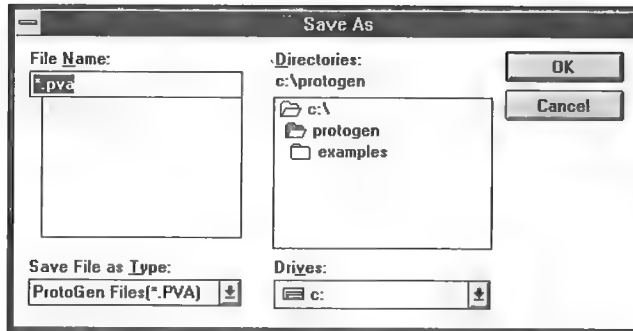


See the description of ProtoGen's main menu File | Open in Chapter 3. It works the same way.

Click File | Save to save a file under its current name, or click SaveAs to rename the file using the following dialog box:

Figure 4.5  
Save As

Click on the drive and file name you want, or type the new file name in the window under File Name.



See the description of ProtoGen's main menu File | Save and File | SaveAs in Chapter 3. They work the same way here.

Click on File | Done to exit the Menu Designer.

---

## Edit

The Edit menu lets you cut, copy, and paste one line at a time from the list box, which is the large box in the center of the Menu Designer. The cut, copy, and paste operations work as you'd expect. The Windows Clipboard holds data for the cut and paste operations.

- Cut cuts the highlighted line from the list box, and puts it in the Clipboard.
- Copy puts a copy of the highlighted line into the Clipboard.
- Paste inserts the contents of the Clipboard at the current position in the list box.

You can't copy and paste popup or cascaded menus in their entirety. Attempting to cut cascaded menus causes ProtoGen to display a warning message.

**Important** Cutting the last menu item in a popup changes the popup to a menu item.

---

## Menu

Menu contains two selections that help you design menu templates for storage in .MNU files. The selections on Menu, *Add Menu Item* and *Add Menu Item To Action Bar*, insert canned, template text ("MenuItem" and "Popup") into the Menu Designer list box. These selections help you position the canned text at the end of a menu-level or menu bar during creation of a menu

template. You change the canned text to actual menu and menu item names when you design the actual application menu from the template.

**Add Menu Item** Add Menu Item appends the text “MenuItem” to the end of whichever menu section is focused in the list box. The Menu Designer Append button performs like Add Menu Item, except the button puts text you type in the single-line entry box at the end of the focused menu section of the list box.


For example, assume the Menu Designer list box contains an .RC file represented in the following way:

```
&File >>
    &Load
    &Save
    &Print
    SEPARATOR
    E&xit
&Setup >>
    &Games
    &Stats
    &Week >>
        Next
        Prev
&Pick
    &Winners
&Display >>
    &Picks
    &Ratings
    &Help >>
```

Append “MenuItem” to the &Setup menu by focusing on &Games, &Stats, or &Week. Select Menu | Add Menu Item.

ProtoGen inserts “MenuItem” in the following location:

```
&File >>
    &Load
    &Save
    &Print
    SEPARATOR
    E&xit
&Setup >>
    &Games
    &Stats
```


 &Week >>  
 Next  
 Prev  
 MenuItem  
 &Pick  
 &Winners  
 &Display >>  
 &Picks  
 &Ratings  
 &Help >>

## Add Menu Item to Action Bar

*Cascading menus are often called "popup" menus, but use of the text "Popup" by Add Menu Item doesn't mean the menu is cascading.*

The Add Menu Item to Action Bar selection on Menu appends a main menu called "Popup" to the end of the data displayed in the list box. "Popup" appears in the rightmost position on the Menu Designer's prototype menu above the list box. This feature is handy when you want to add a new menu and ensure it's placed at the end of the prototype menu bar. Later, use the Cascade button on the menu labeled "Popup" to make it cascading.

## Menu Designer buttons

The Cut, Copy, and Paste buttons behave exactly the same as the equivalent menu items in the Edit menu.

- Append adds .RC or .MNU entries from the single-line entry field above the listbox to the end of the current level.
- Insert adds a menu item just previous to the currently highlighted menu item in the listbox. The name of the menu item is determined by the contents of the single-line entry field above the listbox.
- Change puts the contents of the single-line entry field in the highlighted location of listbox.
- Cascade converts a menu item into a cascading menu item, commonly called a popup. This button is ineffective if used when highlighting a menu that's already cascaded.

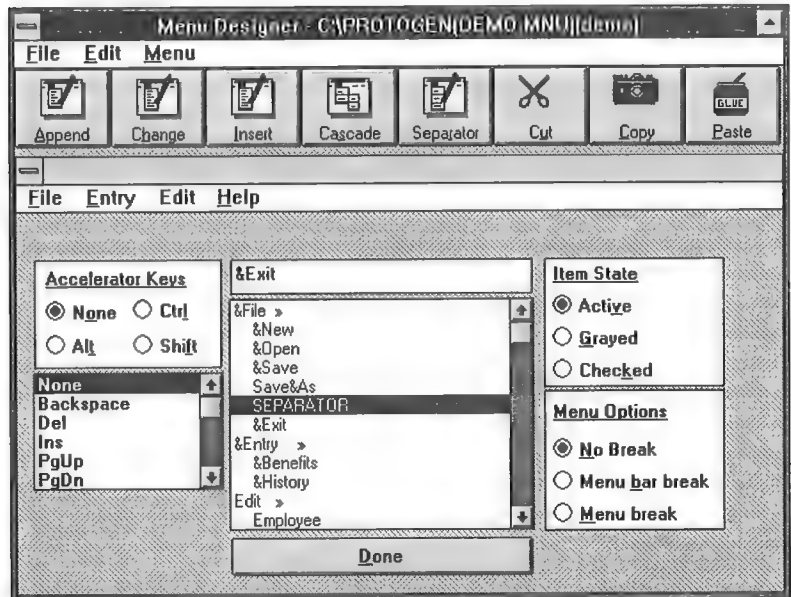
If fewer than three menu items are needed, it's easy to delete one. If more than three menu items are needed, select a menu item within a popup menu, and click the Append button to add them. Use the Cut button to delete extra menu items. Cascaded menu items serve as headings for child menus. A cascaded menu has an arrow on the right side pointing to the right. The arrow indicates that selecting the item brings up another menu.

Items in a cascaded menu are usually closely related functionally.

Cascaded menu items can't be cut and pasted with the Clipboard facility.

- Separator adds the word "SEPARATOR" above the focused menu item in the list box, as shown in the following figure:

Figure 4.6  
Separator function



Separators can't be added to cascaded menu items.

When you popup the prototype menu, a separator line is displayed at the indicated location.

## Examples

For example, assume you've created a popup menu, called File, with menu items Open, Save, and Exit. Begin adding ampersands by selecting the File popup menu in the Menu Designer list box.

### Change example

Double-click on the list box or press *ENTER*. Add the ampersand character in front of the F in the word File. Press *ENTER* and the ampersand appears in the list box. Repeat this process for Open, Save, and Exit by clicking on each item in turn and editing its text in the edit box.



### Insert example

Assuming you have two popups, File and Help, created and displayed on the prototype menu bar, insert a popup called "Edit" between them. Select the Help menu item from the list box and press Enter. Change "Help" to "Edit" and click the Insert button to insert "Edit" in front of the Help.

### Cascade example

To create a cascaded menu or popup menu, select the menu item to be cascaded from the list box. Click the Cascade button to insert three menu items under the focused menu. The Menu Designer names the menu items Item1, Item2, and Item3. Rename the three menu items to suit the application, for example, rename Item1, Employee and place the ampersand character in front of the letter E. Change Item2 into the word Department placing the ampersand mnemonic character in front of the letter D. Change Item3 into the word Location placing the ampersand mnemonic character in front of the letter L. Inspect the menu items under the popups in the main action bar to verify the changes.

### Separator example

To add a menu separator to a popup menu, select the menu item that you want to follow the separator. Click the Separator button. The word "SEPARATOR" appears immediately below the selected menu item and a separating line is displayed at the appropriate place on the prototype menu.

## Item State

---

*Text entered in the single-line entry field takes effect when you press the Change button or ENTER, whereas the option selected for the menu item is effective immediately.*

The Item State refers to the initial state of the menu item that can be changed by ProtoGen. The Item State controls to the right of the Menu Designer list box, Active, Grayed, and Checked, are located on the right side of the list box. The default item state is active, which means that the focused menu item invokes a function or dialog box. Select a radio button to make menu items grayed or checked. Grayed items don't invoke a function or dialog box. Checked items are displayed on the menu with a check mark and indicate an "on" state.

## Menu Options

---

The Menu Options—No break, Menu bar break, and Menu break—control space on the left-hand side of the menu items on the main menu bar. Menu bar break displays the focused menu item with a breaking line on the left-hand side of the menu, while Menu break displays the extra space without the line.

## Accelerator keys

---

Menu accelerator keys let the user quickly make menu choices through the keyboard rather than the mouse. Accelerator keys for the most frequently performed operations in an application should be easy-to-remember and easy-to-type.

Accelerator keys consist of one or two keys that select a menu item. Accelerators can't be added to popup or cascaded menus, and they don't work in test mode. A menu accelerator can be associated with a menu item by performing the following steps:

1. Select the menu item from either the list box or from the popup on the prototype menu bar.
2. Choose one of the radio buttons (None, Ctrl, Alt, or Shift) to the left of the menu list box.
3. Select the second key, *Backspace*, *Del*, *Ins*, or one of the other keys listed in the scroll box below the Accelerator Keys.
4. Inspect the popup menu on the prototype menu that contains the menu item selected in step 1. Seeing the names of the accelerator keys on the prototype menu verifies that you set them correctly.

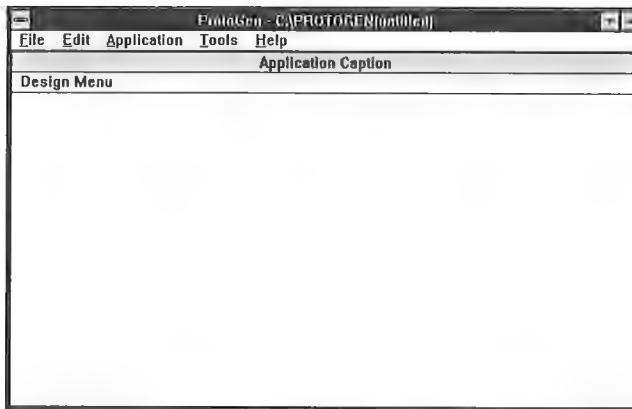
## Exiting Menu Design

---

Select File | Done from the menu bar to exit the Menu Designer. The ProtoGen main window is displayed. When you exit the Menu Designer, the following screen is redisplayed:

Figure 4.7  
Application Generator

*Click on the Design Menu or whatever you've named your menu, which appears under Application Caption, to bring up the controls for linking dialog boxes to the menu.*



Attach dialog boxes to the menu items or attach a dialog box to the main window as described in the next chapter.

## Main Window Dialog

---

The Edit | Main Window Dialog menu item lets you select a pre-existing dialog resource for display in the client area of your application and control the flow of screens in your Windows application. Dialog boxes and data entry screens can be linked to menu items and to buttons and bitmaps on other windows easily.

See the next chapter for instructions on how to link dialog boxes to menus or to the application's main window.

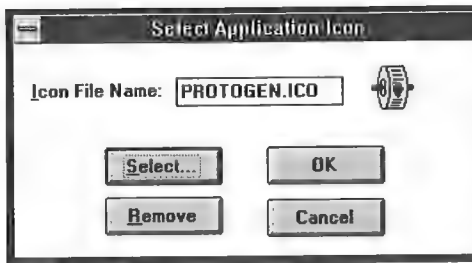
## Application Icon

---

The application icon, caption, and background color of the application's main window can be set either before or after using the Menu Designer.

To select an icon for the application, select Edit | Application Icon from the ProtoGen main menu. The following dialog box is displayed:

Figure 4.8  
Select Application Icon



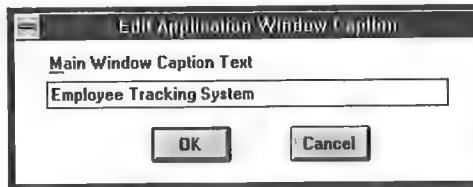
Click the Select button. ProtoGen displays a list of icons. If you select an icon from another directory, ProtoGen asks if you want the icon file to be copied to the working directory and made available to the resource compiler. The application's icon is displayed when the compiled application is minimized. ProtoGen won't display the icon in test mode or while the application is being prototyped.

## Window Caption

---

To set the application window caption, choose Edit | Window Caption from the main menu and enter the caption text in the following dialog box:

Figure 4.9  
Edit Application Window  
Caption



Click the OK button to see the caption displayed. An alternate way of editing the application caption is to click on the caption bar to display the Edit Application Window Caption dialog box.

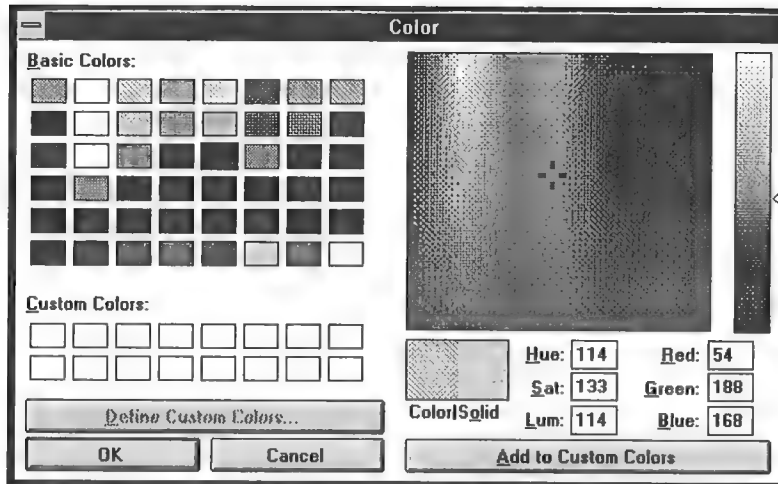
## Window Colors

---

To select the background color of the application's main window, choose Edit | Windows Colors from the main menu. (If you're in Edit | Main Window Dialog mode, exit to select background colors.) The following dialog box lets you choose one of 64 predefined colors or one of 16 custom-defined colors:

Figure 4.10  
Color selection

*Double-clicking on the application client area brings up the color selection dialog box, also.*



To define a custom color, click the Define Custom Color push button, and select a color from the color selection controls. Click on the Add to Custom Colors button. The custom color choice is added on the left of the dialog box.

Select the desired color for the application background, and click the OK button. The color is applied.

When you save a color palette, it's only saved for the current session.

## Remove Window Colors

---

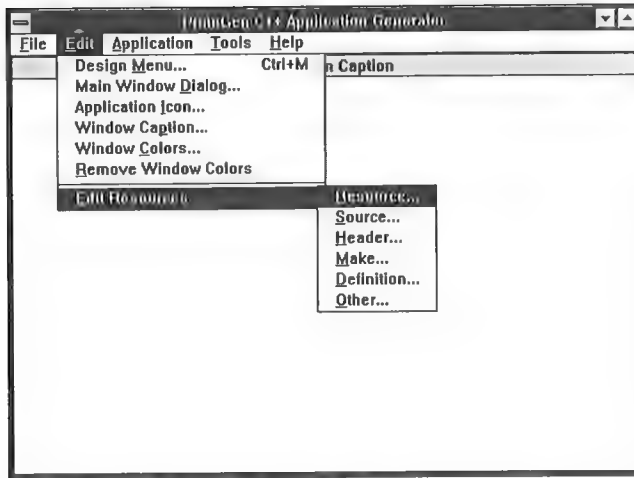
Selecting Edit | Remove Window Colors removes the color selected with Edit | Window Colors.

## Edit Resources

---

Edit | Edit Resources displays the following edit menu:

Figure 4.11  
Edit Application Window  
Caption



Edit | Edit Resources works with files less than 32K. It contains the following menu items:

- Resource lets you bring up a .RC file for editing.
- Source lets you edit .C or .CPP files.
- Header lets you select .H files for editing.
- Make lets you select .MAK files for editing.
- Definition lets you select .DEF files for editing.
- Other brings up a blank, untitled window, which can be used to edit text files.



## Dialog boxes

Dialog boxes are an important aspect of most business application development. *Dialog box* is a generic term for many kinds of input screens used in graphical user interface (GUI) environments. The main window of an application might actually be a dialog box, such as the Windows PIF Editor. A series of dialog boxes might cover the functionality of an entire application. In general, however, a dialog box is a window that contains entry fields and controls. You don't have to link a dialog box to every menu item. Some menu items perform some function or bring up a submenu. When a dialog is attached to a menu item, the menu item should be displayed with an ellipsis, "...". The Open menu item on ProtoGen's File menu is an example of such a menu item.

This chapter discusses how to link dialog boxes to the application's main window and to menus and push buttons.

### DLG files

---

.DLG files, like .RC files, are text files. By convention .DLG files contain only dialog resources, although it's possible for a .DLG file to hold other resources. .RC files usually contain various resources. ProtoGen assumes that .DLG files already exist in the working directory. If you don't have .DLG files, create them with the Windows SDK or with another tool, such as Borland's Resource Workshop.



The following steps create a .DLG file with Resource Workshop:

1. Invoke Resource Workshop.
2. Select File | New project.
3. Set the program file type equal to RC.  
Resource Workshop displays an untitled RC dialog box.
4. Select Resource | New
5. Set the resource type equal to dialog.  
Resource Workshop displays a dialog editing box.
6. Create the dialog box.
7. Save the file as a .DLG file by setting file type equal to DLG.



If you change a .DLG file in Resource Workshop or with another tool after linking it to a ProtoGen application, the link is invalidated and error messages are displayed when you try to generate or compile the application.

Although it's unusual to have a .RC file that contains definitions of dialog boxes, if you have such a file, copy it and rename it with a .DLG extension. With a .DLG extension, ProtoGen can access dialog box definitions and link the dialog boxes to menu items.

ProtoGen helps you link dialog boxes described in .DLG files to both menus and push buttons.

## Linking dialog boxes to menu items

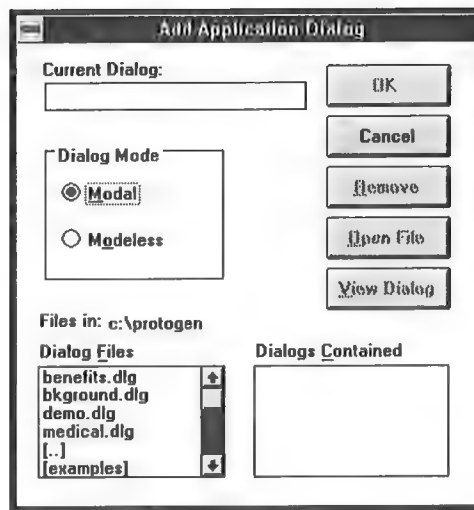
---

Perform the following steps to link a dialog box to a menu item:

1. Exit the Menu Designer to display the main ProtoGen screen.
2. Select a menu item from a prototype menu (Design Menu in previous figures), which is displayed under "Application Caption".

The following Add Application Dialog screen is displayed if a dialog box isn't already linked to the selected menu item.

Figure 5.1  
Add Application Dialog



The Add Application Dialog buttons perform the following functions:

- Remove isn't applicable. See Edit Application Dialog.
- Open File displays dialog resources contained in a .DLG file. After selecting a file, DIALOG templates contained in the file are listed in the lower right window for selection.
- View Dialog displays the graphic dialog box represented by text in the .DLG file.
- Dialog Mode sets the dialog box to be either modal or modeless.

Radio buttons Modal and Modeless make the dialog box either modal or modeless.

- Most dialog boxes are modal, which means the user can't access other parts of the application until the dialog is finished. The modal dialog almost always offers at least two choices, OK or Cancel, for terminating the window.
- A modeless dialog lets the user switch to other applications that are running. A modeless dialog doesn't limit the user to other parts of the application. A modeless dialog resembles a popup window, except it contains entry fields and controls that gather or display information.

## One .DLG linked to multiple menus

### Step-by-step linking

You can't link a .DLG file to more than one menu item unless you rename the file, creating a unique name for each additional link.

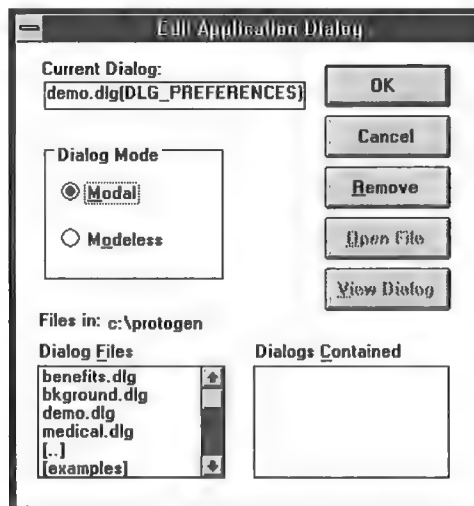
To link dialog boxes in the .DLG files to a menu item, perform the following steps:

1. Select Open File from the Add Application Dialog.
2. Select the .DLG file you want to link to the menu item from the Dialog Files window of the Add Application Dialog Dialog box. The Dialogs Contained window shows contents of the .DLG file.
3. Click the View button to inspect the contents of the .DLG file.
4. Change the modality of the dialog box by selecting the dialog mode to be modal or modeless.
5. Click OK to link the file.

## Unlinking dialog boxes from menu items

To display the following Edit Application Dialog box press *F2* or right-click the mouse when a dialog box is onscreen.

Figure 5.2  
Edit Application Dialog



## Remove

---

The buttons and list boxes on the Edit Application Dialog are like those on the Add Application Dialog, previously described, except Remove unlinks a menu item in a .DLG file from the application.

To remove a link to a dialog box from an application, perform the following steps:

1. Click on the menu item that the dialog box is attached to.
2. Navigate to the dialog box by clicking on the menu items and the buttons containing subdialog boxes, until ProtoGen displays the dialog box you want to remove.
3. Right click the mouse or press the *F2* function key to bring up the Edit Application Dialog screen.
4. Click on Remove and click OK to confirm.

The dialog box is unlinked.

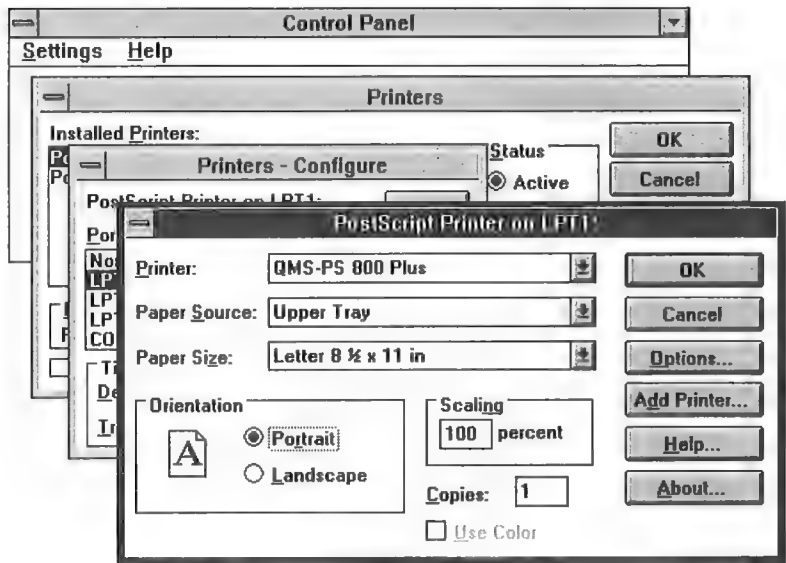
## Linking nested dialog boxes

---

A nested dialog or subdialog is a modal dialog initiated from a parent dialog. The nested dialog gets information related to an element of the parent dialog. The nested dialog might call a second or third dialog to get more information or display data. ProtoGen lets you link a long tree of child dialog boxes. The following figure shows a nested dialog:

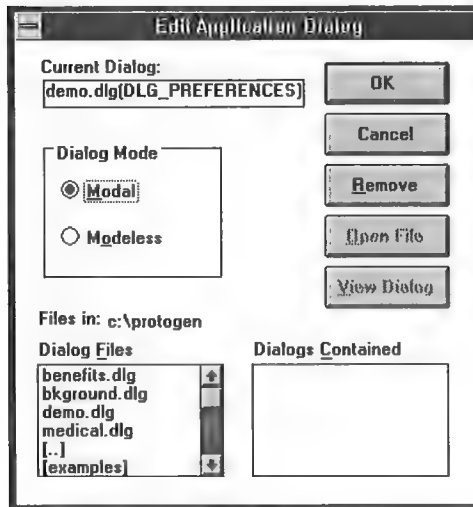
Figure 5.3  
Nested dialog boxes

*ProtoGen lets you attach subdialogs to push button controls on parent dialog boxes.*



Subdialogs are attached by clicking on push buttons within other dialog boxes, and following the steps for linking a dialog box to a menu item.

Figure 5.4  
Edit Application Dialog



ProtoGen displays the Edit Application Dialog screen from a previously linked dialog box or from the Add Application Dialog screen when you press *F2* or when you right click the mouse.

If a dialog box is already linked to the menu item, its name is displayed in the Current Dialog window.

## Escaping from dialog box add or edit mode

---

Sometimes it's hard to remember how to escape from the dialog box editing mode. After you select a noncascaded menu, get the Add Application Dialog on the screen, and fill it in. The linked dialog box is displayed, usually with OK or CANCEL buttons.



Clicking OK or CANCEL on the prototype tells ProtoGen you want to add another dialog box instead of putting away the linked dialog box. Double-click on the client area of the user-defined dialog box to exit add or edit dialog box mode.

## Linking dialog boxes to the main window

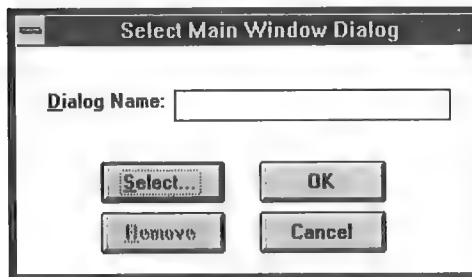
---

The main ProtoGen window's Edit menu selection, Main Window Dialog, takes dialog boxes you've designed and puts them in the main window of your application. When the standard Windows main menu isn't suitable for an application, a dialog box that fulfills the special requirements can be linked as a substitute for the Windows main menu.

To add a dialog box to the main window of your application, perform the following steps:

1. Select Edit | Main Window Dialog. The following dialog box is displayed:

Figure 5.5  
Main Window Dialog



2. Click the Select button to display Add Application Dialog if a dialog box isn't linked, or right-click to display the Edit Application dialog if one is already linked.

Choose a dialog file from a list of .DLG files in the current working directory, navigate to other directories and select a file, or type in the name of the file in the Dialog Name window.

3. Press the OK button.

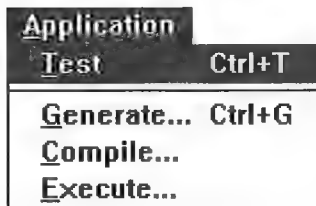
If a dialog box is attached to the application main window, the file name appears in the Dialog Name window. When the Select button is chosen, .DLG files located in other directories are copied to the working directory.

4. Click OK to link the dialog box in the .DLG file to the application main window, or click Remove to unlink the dialog box from the window.

## *Application menu*

This chapter discusses menu selections on the following Application menu, which lets you test and turn your prototype application into source code.

Figure 6.1  
Application menu



The Application menu items perform the following functions:

- Test lets you exercise your application interactively.
- Generate creates source code for your application.
- Compile prepares and links code, producing an executable file with symbolic debugging codes if specified.
- Execute runs the program under DOS.

## Test

---

To get the look and feel of an application, ProtoGen's prototype is run in test mode. To place the application in test mode, choose Select Application | Test. The ProtoGen main window and main



menu disappear, displaying the application alone. ProtoGen displays dialog and subdialog boxes when appropriate menu items are selected, but the application icon isn't displayed in test mode. Testing your application in this manner ensures that the screen flow accomplishes your objectives.

Either right-click the mouse or press *F4* to terminate testing. The ProtoGen main window and main menu reappear.

Accelerator keys don't work in test mode.

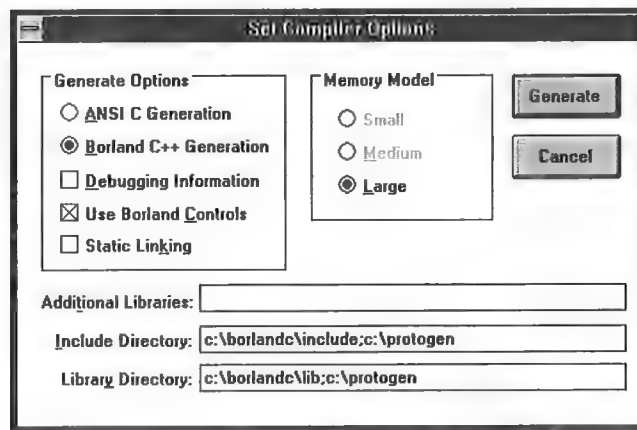
## Generate

Before attempting to generate code, check File | Header and ensure that ProtoGen knows where to find the application header file.

Selecting Application | Generate displays the following dialog box:

Figure 6.2  
Set Compiler Options

*Static Linking is grayed out in ANSI mode.*



For use with Borland C++, enter the following include directories to be searched. Type the semicolons, but not the carriage returns:

- *Drive:* \BORLANDC\OWL\INCLUDE;
- *Drive:* \BORLANDC\CLASSLIB\INCLUDE;
- *Drive:* \BORLANDC\INCLUDE



Enter the following lib directories to be searched:

- *Drive:* \BORLANDC\OWL\LIB;
- *Drive:* \BORLANDC\CLASSLIB\LIB;

## Generate Options

The functions performed by the Generate Options window are described in the following sections.

### ANSI C Generation

This option creates ANSI C standard source code. One primary application source file is generated that takes the name of the application itself. This module contains the **WinMain** function and the window procedure for the application's main window.

The main window procedure contains function calls for each of the dialog boxes attached to menu items. The dialog functions reside in separate source files, named for the dialog they display, and they're linked together in the link file.

Any subdialog boxes attached are generated as individual source modules called from the window procedure of the dialog they're attached to.

Header information for the menu #define preprocessor directives and all dialog and subdialog boxes is stored in the application header file.

If you're using Turbo C++ for Windows, you can't use the compile option to compile the application directly from ProtoGen, because Turbo C++ doesn't include a command-line compiler or resource compiler. To compile your application, you must take the following steps:

1. Select Application | Generate to create the necessary source code, header, resource, and project files.
2. Start Resource Workshop, open your application's .RC resource file, and save it as a .RES file.
3. Start Turbo C++ for Windows, open your applications project (.PRJ) file, and select Make from the Compile menu. Your application is compiled.

### Borland C++ Generation

This option generates code for the Borland ObjectWindows C++ library.

### Debugging Information

This option places the necessary switches into the application makefile to include debugging information in the executable file.

After finishing the prototype of your application, ProtoGen generates the code for it. The generated code includes a main source module (APPNAME.C or APPNAME.CPP) for the application's main window, a source module (DLGNAME.C) for each dialog that is linked to the application if ANSI C code has been generated, a header file (.H) containing field #define preprocessor directives and variables where these have been provided, a resource file (.RC) a definition file (.DEF), a makefile (.MAK), and a link file (.LNK) file. These files comprise the complete source and support files necessary to build the application into a stand-alone executable.

- Use Borland Controls    The Borland Controls check box generates code to initialize and load the Borland dynamic link library, called BWCC.DLL. This DLL contains custom controls that give Borland dialog boxes and screens a sophisticated three-dimensional appearance.
- Static Linking    If you're generating code for ObjectWindows, statically link it by clicking the Static Linking check box, which creates an executable file that contains the ObjectWindows library as opposed to a standard application working in conjunction with the ObjectWindows DLL. If you're generating code with the Borland ObjectWindows library without static linking, you can generate code only for the Large memory model. ProtoGen selectively enables and disables some of these options according to the type of code generation specified.
- Once you choose the options from the Generate Options window, click the Generate button. ProtoGen outputs code to the various files needed for compiling programs. The modules are structured in the following way:
- C++ code for ObjectWindows is generated as a single .CPP source file. Application class and main window class declarations appear first, followed by class and constructor declarations for each dialog box in the application.
  - The application class is a descendant of **TApplication**. The main window is a descendant of **TWindow** unless the main window is a dialog, in which case it is declared as a descendant of **TDialog**. Each dialog class is a descendant of **TDialog**. Message response member functions are declared for each message that invokes a dialog or subdialog.

- Next the member functions for the dialog boxes themselves are generated. The purpose of the dialog member functions is to invoke subdialog boxes attached via the dynamic dispatch table routing messages.
- Other member functions can be added to the dialog classes by inserting code between the regeneration brackets provided. The following excerpt from source code shows an example of these brackets:

```
class TBENEFITSDlg : public TDialog
{
public:
    TBENEFITSDlg(PtWindowsObject AParent, LPSTR AName);
    virtual void Ok(RTMessage Msg) = [ID_FIRST + IDOK];
    //REGEN_BENEFITS_ROUTING
    //REGEN_BENEFITS_ROUTING
};
```

The brackets are //REGEN\_BENEFITS\_ROUTING.

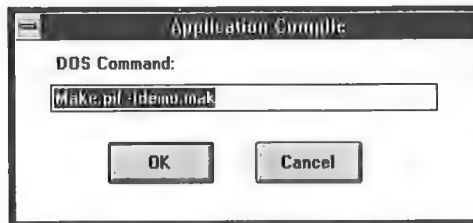
## Compile

---

Once the menu and screen flow has been set up and the code generated, compile the resulting application from within ProtoGen by performing the following steps:

1. Click on Application | Compile from the ProtoGen menu to display the following dialog box:

Figure 6.3  
Compile dialog box



2. Enter the DOS command for compiling in your environment.
3. Click the OK button.

ProtoGen submits the command line to DOS for execution. If your system doesn't have enough memory or if your Windows PIF Editor settings for COMMAND.COM are not set high enough for

memory requirements, you might not be able to load and complete the compilation. In general, 2-3MB is the amount of memory needed to compile an application. If you don't have the necessary memory, exit Windows to carry out the compilation process.

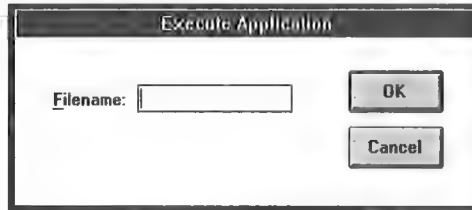
## Execute

---

Once the example application has been compiled, run it from within ProtoGen by performing the following steps:

1. Choose Application | Run from the ProtoGen main menu. The following dialog box is displayed:

Figure 6.4  
Execute dialog box



2. Type in the file name to run.
3. Click the OK button to execute the application.

ProtoGen minimizes itself while the application runs.

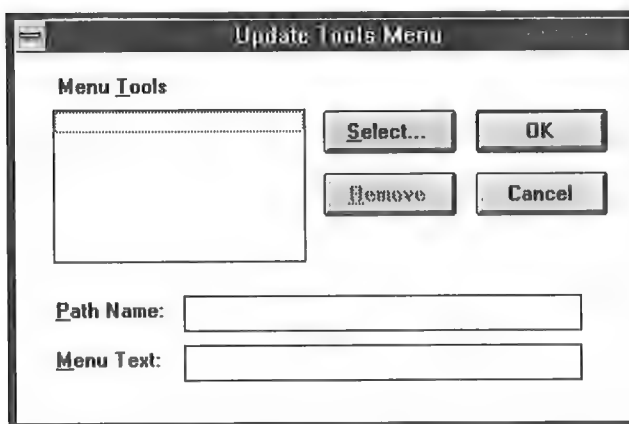
## Tools and other topics

This chapter covers the Tools menu and miscellaneous topics.

Selecting Tools from ProtoGen's main menu displays one menu item, Define Tool. Tools lets you shell out to a selected Windows program from ProtoGen.

Selecting Define Tool displays the following dialog box:

Figure 7.1  
Update Tools Menu



Current programs on the Tools menu are displayed in the Menu Tools window. To add or delete programs from the Tools menu, fill in the dialog box according to the following procedure:

1. Enter the full (absolute) path and name of a Windows program, such as Borland's Resource Workshop, in the Path

- Name window either by typing it or by pressing Select to see a list of windows executables and choosing one of the files.
2. If you're adding instead of removing a program to the Tools menu, enter a title to be displayed on ProtoGen's Tool menu as a menu item for shelling out to the program entered in step 1. If you're removing a program, use the Select button to select it, or type in the path and name for the program.
  3. Click OK to add the program to the Tools list or Remove to remove it.

## Hints on using ProtoGen

---

ProtoGen lets you make the following kinds of modifications to an application:

- application flow
- link dialog boxes created previously with a resource editor
- changes to the application source code directly

To alter application flow add, insert, or delete menu items and dialog boxes to the application by opening the .PVA in ProtoGen, interactively performing redesign work, and rebuilding the application source code.

If you only want to make a change to a single dialog, you can use Borland's Resource Workshop or ProtoView's ViewPaint to edit .DLG files, then recompile the application resources.

The third type of modification involves making source code changes. These changes can occur anywhere within the specially placed code-regeneration brackets provided in the source files. When code is placed in these areas it is preserved by ProtoGen from one code generation to the next.

Although .PVA files are in ASCII format and can be edited, it is best to make changes only through ProtoGen.

## A

- accelerator key
  - assigning from Menu Designer 27, 28, 35
  - defined 6
  - documentation of 7
  - for ProtoGen operation 6, 8
- Active button on Item State 34
- Add Application Dialog 42
- Add Menu Item 31-32
- Add Menu Item to Action Bar 32
- ANSI C code generation 51
- Append 32
- Application Caption 7, 20
- Application Icon 36
- Application menu 49-54
  - Compile on 49, 53-54
  - Execute on 49, 54
  - Generate on 49, 50-53
  - illustration of 8
  - Test on 49
- Application Prototype 7
- AUTOEXEC.BAT file 5

## B

- Borland C++ files 50
- Borland's Resource Workshop 41-42, 51, 56
- Borland's Turbo C++ for Windows 51
- BWCC.DLL file 52

## C

- .C or .CPP file 52
- CANCEL 7
- capabilities of ProtoGen 1, 2, 56
- Cascade button 32, 34
- cascaded menu limitations 33, 35
- Change button 32, 33
- Checked button on Item State 34
- color of application screen changing 37-38

- Compile on Application menu 49, 53-54
- Copy 30, 32
- Cut 30, 32

## D

- .DEF file
  - editing of 39
  - generation of 52
- Definition on Edit | Edit Resources 39
- Design Menu 25-36
- "Design Menu" on menu bar 26
- dialog box
  - adding to main window 47-48
  - adding to menu item 42-44
  - linked to main window 47-48
  - linking to menu item 44, 42-44
  - nested 45-47
  - unlinking 44-45
- dialog boxes defined 41
- Dialog Mode 43
- .DLG file
  - creation of 41-42
  - defined 41
  - linking to multiple menu items 44
- Done
  - on Menu Designer File menu 30
  - on Menu Designer menu 35

## E

- Edit Application Dialog 44, 47
- Edit menu
  - main
    - Application Icon on 36
    - Design Menu on 25-36
    - Edit Resources on 38
    - illustration of 8
    - Main Window Dialog on 36
    - Window Caption 37



- Window Colors 37-38
- Edit menu on Menu Designer menu bar 30
- Edit Resources 38
- editing files 39
- Execute on Application menu 49, 54
- Exit on main File menu 23
- exiting
  - add/edit dialog box mode 47
  - from main menu 23
  - from Menu Designer 30

## F

- features of ProtoGen 1, 2, 56
- File menu
  - main 19
    - described 8
  - Exit on 23
  - Header File on 22-23
  - New on 19
  - Open on 19
  - Working Directory on 21
- Menu Designer
  - Done 35
  - Done on 30
  - New on 26, 27, 29
  - Open on 27, 29
  - Save on 28, 29
  - SaveAs on 29

## G

- Generate on Application menu 49, 50-53
- Grayed button on Item State 34
- grayed-out menu items 6

## H

- .H file
  - editing of 39
  - example of 22-23
  - for Borland C++ 50-51
  - general 51
- hardware and software requirements 2
- Header, on Edit | Edit Resources 39
- header file
  - editing of 39
  - example of 22-23
  - general 51

- required for Borland C++ 50-51
- Header File, on File menu 22-23

## I

- Insert button 32
  - example of 33
- installation 4-5
- Item State 34

## K

- keyboard shortcut
  - assigning from Menu Designer 27, 28, 35
  - defined 6
  - for ProtoGen operation 7, 8

## L

- .LIB file
  - BWCC.DLL 52
  - for Borland C++ 50-51
- library files for Borland C++ 50-51
- link file 52
- linking dialog box
  - to main window 47-48
  - to menu item 42-44
- list box 27, 28
- .LNK file 52

## M

- main menu 7
- Main Window Dialog 36
- .MAK file
  - editing of 39
  - generation of 52
- Make, on Edit | Edit Resources 39
- memory requirements 54
- Menu, on Menu Designer menu 30
- Menu bar break button on Menu Options 34
- Menu break radio button on Menu Options 34
- Menu Designer 25-36
  - illustration 27
  - list box on 27
  - menu bar on 29
  - single-line entry box on 27
- Menu Options 34
- menu templates *See* .MNU file

mnemonic 5, 6

.MNU file

defined 28

general 20-21

mentioned 27

naming of 21

preserving of 21

updating of 23

modal dialog box 45

modal/modeless dialog box 43

## N

nested dialog box 45-47

New

on main File menu 19

on Menu Designer menu 26, 27, 29

No Break button on Menu Options 34

## O

OK button 7

Open

on main File menu 19

on Menu Designer menu 29

Open File button, on Add Application dialog 43

Open on Menu Designer menu 27

Other on Edit | Edit Resources 39

## P

Paste 30, 32

PATH variable 5

popup menu limitations 33

prototype menu 28

.PVA file

contents and saving of 21

general 20-21

selecting of 23

updating of 23, 56

## Q

quit

from main File menu 23

from Menu Designer 35

quitting, add/edit dialog box mode 47

## R

.RC file

contents of 21, 28

editing of 39

example of 28

general 20-21

generation of 52

opening of 27, 29

Remove button

on Add Application dialog 43

on Edit Application Dialog 44

resource script file *See* .RC file

Resource Workshop 41-42, 51, 56

## S

Save

on main File menu 20-21

on Menu Designer File menu 28, 29

SaveAs

on main File menu 21

on Menu Designer File menu 29

Select button on Add Application Dialog 47

Separator button

example of 34

use of 33

shortcut key sequences

assigning from Menu Designer 28, 35

defined 6

for ProtoGen operation 7-8

single-line entry box 27

software and hardware requirements 2

source file 52

static linking 52

## T

Test on Application menu 49

3-D screens 52

toggle to Menu Designer 27

Tools menu 54-56

Turbo C++ for Windows 51

## U

unlinking dialog box 44-45

Update Tools on Tool menu 54-56

## **V**

View Dialog button on Add Application Dialog  
43

## **W**

Window Caption 37  
Window Colors 37-38  
Working Directory on File menu 21

**ProtoView Development Corporation**

353 Georges Road  
Dayton, New Jersey 08810

PN 14MN-PTG01-10